

Social and Ethical Issues Involved in the use of Software

- **Bug Free Software**

A **software bug** is an error, flaw, mistake, failure, or fault in a computer program that prevents it from behaving as intended (e.g., producing an incorrect result).

Most bugs arise from mistakes and errors made by people in either a program's source code or its design

There are many types of bugs, bugs that don't affect the functionality of the program, more serious bugs that cause the software to crash and security bugs.

Causes: Complex System

Solution: Test more, follow good software procedure
(Page 408)

- Software Piracy

To many people software piracy is a matter of 'innocently' sharing software amongst friends and they can't really see anything wrong with it. In fact, there is nothing "innocent" or acceptable about it. It is against the law.

Deliberate counterfeiting is a serious problem. Ten years ago, most copying was done on a small scale, usually amongst friends or work colleagues. Now, it accounts for nearly 40% of all software in existence.

Software piracy in the workplace has also become far more common. Piracy can be as simple as two employees installing the same program on their computers when the license agreement only allows one. It can be as widespread as the deliberate, systematic copying of programs throughout an organization

The Internet has also made it much easier to steal market and distribute copyrighted material.

Pirated software on the Internet is often shared via bulletin boards, e-mail, news groups, site links and Internet auction sites.

Many of you may think, 'what's the problem, the software companies make tons of money anyway and it isn't harming anyone'. Well think again, it is!

I bet you look forward to new games being released so that you can try something new out on your games console. These games cost millions of pounds to develop. Where do you think that money comes from? It comes from sales of games. So, every time you copy a game from a friend, the software company has lost more money to invest in creating new ones for you to play.

It's estimated, in fact, that people breaking the law and downloading copyrighted music costs the industry something like £8.5 billion a year – and loses £1.5 billion for the VAT man.

By just cutting the software piracy rate from 25% to 15% would result in an extra 40,000 jobs in the IT sector and an extra £2.5 billion in tax revenues. Besides being illegal, using pirated software is not a good idea because of the problems it can cause:

- **There is no quality control.** The software could contain a virus, trojan or key logger as a deliberate attempt to provide criminals with a 'back door' to your computer.

- **There will be no warranty or support if things go wrong** - which they often do.

- It may be missing important key functions which could make it less useful. Perhaps it doesn't save properly or maybe it is missing some font sets.

- **Pirated software could possibly expire** in the middle of a project leaving you stranded and unable to get hold of your work or continue doing more.

- **If caught, you could end up in court** with a large fine and possible prison sentence.

Copy protection

Hundreds of different methods have been tried to prevent illegal copying. There is no perfect solution, as some methods are more expensive than others.

- **'Serial number'** needed, usually only printed on the original package. The most common method of software protection, but simple to overcome with 'legal' serial numbers

- **'Activation code'** needed, you have to go on a web site to get a registered code. Now becoming very popular with companies, as you often have to provide personal information as well.

- **'Dongles'**, these are bits of hardware that must be fitted to the computer to allow the software to run. Only practical on the most expensive software costing thousands of pounds.

- **'Region coding'** - you can only play disks bought in certain countries. Region codes are very popular with Hollywood companies, who want to control how their DVD films are sold around the world.

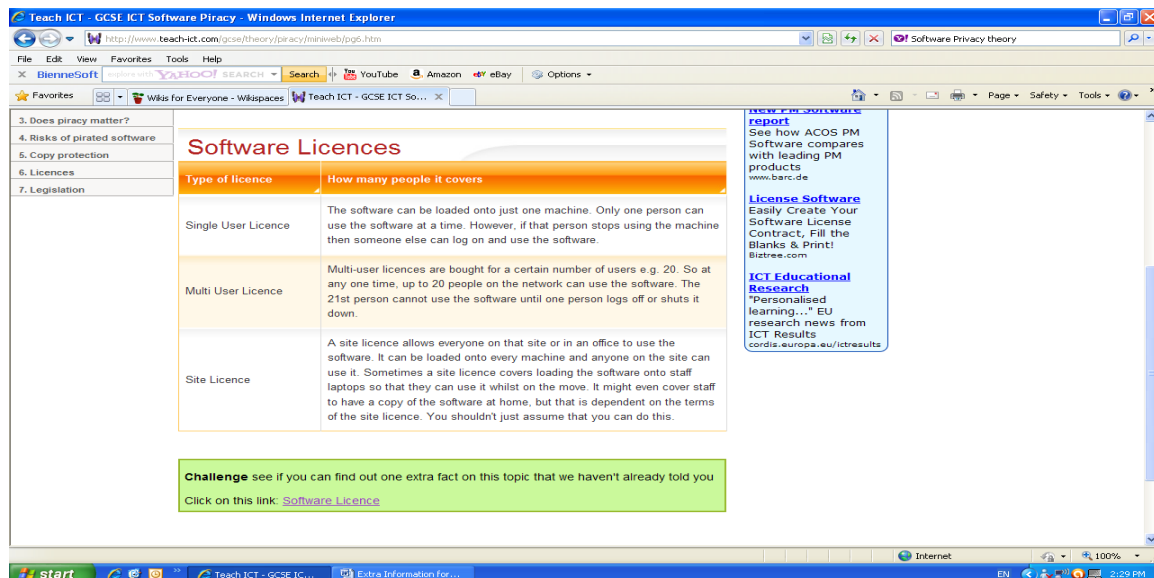
These technology methods are only of limited use as they can all be overcome. The only truly effective way of preventing software piracy is to educate people that it is wrong and harmful to do so.

But saving money is always tempting, and so laws have been created to make it not worth the risk of getting caught. This is especially true for businesses who do not want to harm their reputations and good name.

Licenses

When you purchase a piece of software you don't actually own the software. What you are buying is a license to be able to use the software.

You are able to load your software onto your machine at home and use it for as long as you like. You are not however, allowed to take it to your mates and load it onto his computer - he doesn't have a license to use it



Legislation

The Copyright, Designs and Patents Act (1988) covers the issue of software piracy. This Act is straightforward and simple to understand. It essentially covers three main things:

1. **It is illegal to copy software** without the permission of the rightful owner. Even if you have a licence for the software, you are not the 'rightful owner', you just have permission to use your copy.
2. **It is illegal to actually run** copied (pirated) software.
3. **It is illegal to transmit copied software** over a telecommunications line. You do not have to have a disk or CD to pirate software. Copying through the internet or telephone line is also illegal, as is going to the cinema and recording the latest film on video.



- **Globalization of Software**

So just what is **software globalization**? The short answer is making software products run anywhere. In our context at IBM, it means making applications work seamlessly, regardless of the user's language and culture. Globalization is all about choices: it gives the user the choice to use the original US English version of the product, or in fact, any of the other supported languages.

- Use of password protection to prevent unauthorized access
- Interfaces adapted for the disabled
- Language independence of G.U.I. making computers accessible to a very wide range of users including those with special needs and very small children

In this article, Microsoft has launched its latest operating system, Windows Vista. It talks about how much has changed between its beginning, and its debut. This system has taken a long time to make, and it now has many online competitors, like Google. Not only that, but Microsoft publicly announced that the operating system had problems in it, and it was not perfected at the time of the release. The only reason they released it was because they had to reach a quota. This poses a problem with reliability because you spend a lot of money to buy a new operating system, and it should not come with bugs or problems. This is very important because the Operating systems are the fundamentals that allow users to use the computer. Many consumers depend on the OS to make their lives easier and let them manipulate around the computer without being an expert. This also brings up the discussion of whether or not companies should release new products, knowing that there are bugs in the system. Because so many people depend on the OS to work, it shouldn't be right releasing a bugged program just because they need to meet a quota. But because of the pressure put on by businesses, it really leaves the creators not much of a choice: they need to create something newer and faster than their competitors, or else they will not be able to compete. Because of this business debacle, the consumers are the ones stuck with the bugs in the newly released programs.

Bug-free software: Often times software companies will rush out projects with sloppy code in order to save money in the short-term, ultimately causing the company to lose money in the long-run. The problem is that the typical code is typically tens of thousands of line long, far too lengthy to quickly go through to find errors, and few programmers want to take the time to meticulously go

through their code to search for errors. However, in the US alone, over \$59.5 billion is lost due to software errors - that's over 0.6% of the domestic product! Furthermore, software error can cost human lives, such as an error in an airplanes flight path causing it to fly at the wrong speed, crash, and kill dozens of flyers onboard. Companies address these issues by making their computer coders to take breaks from coding, and afterwards look back over it to ensure everything is in working order, before they begin meticulous bug testing. Such tends to lead to less buggy coding, which benefits for the consumer and the company